Dimensionality Reduction by Random Mapping: Fast Similarity Computation for Clustering

Samuel Kaski

Helsinki University of Technology, Neural Networks Research Centre
P.O. Box 2200, FIN-02015 HUT, Finland
samuel.kaski@hut.fi

Abstract

When the data vectors are high-dimensional it is computationally infeasible to use data analysis or pattern recognition algorithms which repeatedly compute similarities or distances in the original data space. It is therefore necessary to reduce the dimensionality before, for example, clustering the data. If the dimensionality is very high, like in the WEBSOM method which organizes textual document collections on a Self-Organizing Map, then even the commonly used dimensionality reduction methods like the principal component analysis may be too costly. It will be demonstrated that the document classification accuracy obtained after the dimensionality has been reduced using a random mapping method will be almost as good as the original accuracy if the final dimensionality is sufficiently large (about 100 out of 6000). In fact, it can be shown that the inner product (similarity) between the mapped vectors follows closely the inner product of the original vectors.

1. Introduction

There exists a wealth of alternative methods for reducing the dimensionality of the data, ranging from different feature extraction methods to multidimensional scaling. The feature extraction methods are often tailored according to the nature of the data, and therefore they are not generally applicable, for example, in all data mining tasks. The multidimensional scaling methods, on the other hand, are computationally costly already on their own, and if the dimensionality of the original data vectors is very high it is infeasible to use even linear multidimensional scaling methods (principal component analysis) for dimensionality reduction.

A new rapid dimensionality reduction method is needed for situations where it is impossible to use the original vectors as such and the existing dimensionality reduction methods are too costly. The random mapping method presented in this paper provides a computationally feasible method for reducing the dimensionality of the data so that the mutual similarities between the data vectors are approximately preserved.

The motivation for the method dates back to experiments made by Ritter and Kohonen [10]. They organized words based on information of the contexts in which they tend to occur. The dimensionality of the representations of the contexts was reduced by replacing each dimension of the original space by a *random direction* in a smaller-dimensional space.

It may seem surprising that random mapping can reduce the dimensionality of the data in a manner that preserves enough structure of the original data set to be useful. The main goal of this paper is therefore to explain why the random mapping method works well in high-dimensional spaces, using both analytical and empirical evidence.

2. Random mapping method

In the (linear) random mapping method the original data vector, denoted by $\mathbf{n} \in \mathbb{R}^N$, is multiplied by a random matrix R. The mapping

$$\mathbf{x} = R\mathbf{n} \tag{1}$$

results in a reduced-dimensional vector $\mathbf{x} \in \mathbb{R}^d$. The matrix R consists of random values and the Euclidean length of each column has been normalized to unity.

One way of interpreting the random mapping is to consider what happens to each of the dimensions of the original space $\mathbb{R}^N$ in the mapping. If the ith column of R is denoted by $\mathbf{r}_i$ the random mapping operation (1) can be expressed as

$$\mathbf{x} = \sum_i n_i \mathbf{r}_i \,. \tag{2}$$

This work was supported by the Academy of Finland.

0-7803-4859-1/98 $10.00©1998 IEEE

Here the ith component of $\mathbf{n}$ is denoted by n_i. In the original vector $\mathbf{n}$ the components n_i are weights of *orthogonal* unit vectors, whereas in the expression (2) each dimension i of the original data space has been replaced by a random, non-orthogonal direction $\mathbf{r}_i$ in the reduced-dimensional space.

3. Properties of the random mapping

The utility of the random mapping method for clustering depends fundamentally on how it affects the *mutual similarities* of the data vectors. It is clear that the closer the vectors $\mathbf{r}_i$ in (2) are to being orthonormal the better the similarities of the vectors obtained by random mapping correspond to the original similarities. A hint on why even choosing random directions for the vectors $\mathbf{r}_i$ might be useful has been provided by Hecht-Nielsen [3]: there exists a much larger number of almost orthogonal than orthogonal directions in a high-dimensional space. Therefore, in a high-dimensional space even vectors having random directions might be sufficiently close to orthogonal to provide an approximation of a basis.

Below the distortions that the random mapping method causes on the mutual similarities of data vectors will be characterized statistically.

3.1. Transformation of the similarities

The cosine of the angle between two vectors is a commonly used measure of their similarity. The results in this paper will be restricted to vectors with unit length; in that case the cosine can be computed as the inner product of the vectors.

The inner product of two vectors, $\mathbf{x}$ and $\mathbf{y}$, that have been obtained by random mapping of the vectors $\mathbf{n}$ and $\mathbf{m}$, respectively, can be expressed using (1) as follows:

$$\mathbf{x}^T\mathbf{y} = \mathbf{n}^T R^T R \mathbf{m} \,. \tag{3}$$

The matrix R^TR can be decomposed into two terms,

$$R^TR = I + \epsilon \,, \tag{4}$$

where

$$\epsilon_{ij} = \mathbf{r}_i^T \mathbf{r}_j \tag{5}$$

for $i \neq j$, and $\epsilon_{ii} = 0$ for all i. The components on the diagonal of R^TR have thus been collected into the identity matrix I in (4). They are always equal to unity since the vectors $\mathbf{r}_i$ have been normalized. The units off the diagonal have been collected into the matrix ϵ. If all the entries in ϵ were equal to zero, i.e., the vectors $\mathbf{r}_i$ and $\mathbf{r}_j$ were orthogonal, the matrix R^TR would be equal to I and the similarities of the documents would be preserved exactly in the random mapping. In practice the entries in ϵ will be small but not equal to zero.

Statistical properties of ϵ. It is possible to analyze the statistical properties of the entries in ϵ if we fix the distribution of the entries in the random mapping matrix R, i.e., the distribution of the components of the column vectors $\mathbf{r}_i$. Assume that the components are initially chosen to be independent, identically and normally distributed (with mean zero), and thereafter the length of all of the $\mathbf{r}_i$ is normalized. The result of this procedure will be that the direction of the $\mathbf{r}_i$ will be distributed uniformly. Then it is evident that

$$E[\epsilon_{ij}] = 0 \tag{6}$$

for all i and j, where E denotes the average over all random choices for the entries of R.

In practice we always use one specific instance of the matrix R, and therefore we need to know more of the distribution of ϵ_{ij} to judge the utility of the random mapping method. It can be proven (cf. Appendix A) that if the dimensionality d of the reduced-dimensional space is large, ϵ_{ij} is approximately normally distributed. The variance, denoted by σ_ϵ^2, can be approximated by

$$\sigma_\epsilon^2 \approx 1/d \,. \tag{7}$$

The distribution of ϵ_{ij} for several dimensionalities has been illustrated in Figure 1. *The matrix R^TR will approximate the identity matrix the better the higher-dimensional the vectors $\mathbf{r}_i$ are.*

Statistical properties of the mutual similarities. Now that we know the distribution of ϵ it is possible to investigate more closely how the similarities of the original vectors are transformed in the random mapping. More specifically, given a pair $\mathbf{n}$ and $\mathbf{m}$ of original data vectors it is possible to derive the distribution of the similarity of the vectors $\mathbf{x}$ and $\mathbf{y}$ obtained by random mapping of $\mathbf{n}$ and $\mathbf{m}$, respectively.

Using equations (3), (4) and (5) the inner product between the mapped vectors can be expressed as

$$\mathbf{x}^T\mathbf{y} = \mathbf{n}^T\mathbf{m} + \sum_{k \neq l} \epsilon_{kl} n_k m_l \,. \tag{8}$$

Denote $\delta = \sum_{k \neq l} \epsilon_{kl} n_k m_l$; this expression is the deviance form the original value of the inner product produced by the random mapping.

The mean of δ is zero since the mean of each term in the sum is zero. It will be shown in Appendix B that the variance of δ, denoted by σ_δ^2, can be expressed as

$$\sigma_\delta^2 = [1 + (\sum_k n_k m_k)^2 - 2\sum_k n_k^2 m_k^2]\sigma_\epsilon^2 \,. \tag{9}$$

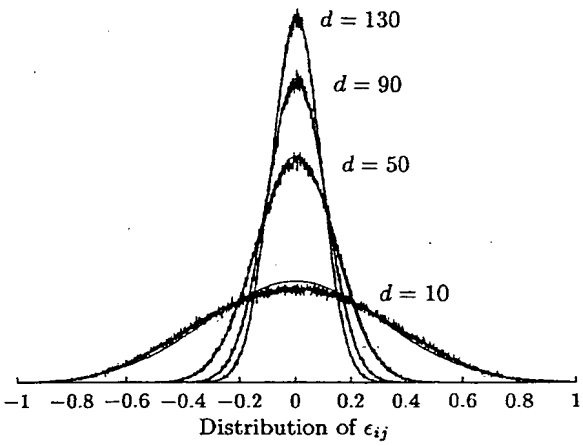

Figure 1. Distribution of the inner products between pairs of random vectors $\mathbf{r}_i$, i.e., the distribution of ϵ_{ij}, for different dimensionalities d. When d increases the inner products become smaller and the vectors $\mathbf{r}_i$ become more orthogonal. The orthogonality will not be perfect but the generally small inner products contribute only small distortions in the similarity computations. The normal distributions with variance equal to $1/d$ are plotted in the figure as well; the curves are only distinguishable from the empirical curves for $d = 10$, which demonstrates that the distribution of ϵ_{ij} approximates the normal distribution already for fairly small values of d.

When the length of the original data vectors $\mathbf{n}$ and $\mathbf{m}$ is fixed to unity their inner product is at most 1 and, based on (7),

$$\sigma_\delta^2 \leq 2\sigma_\epsilon^2 \approx 2/d\,. \tag{10}$$

In summary, the distortion of the inner products produced by the random mapping is zero on the average, and its variance is at most the inverse of the dimensionality of the reduced space (multiplied by 2).

Consider then a simple but instructive setting where the data vectors are constrained to have only a certain amount, L, of ones and the rest of the components are zero. If K of the ones occur in the same position in both of the vectors (i.e., the inner product is K/L) then

$$\sigma_\delta^2 = [1 + (\frac{K}{L})^2 - 2\frac{K}{L^4}]\sigma_\epsilon^2\,.$$

If the inner product K/L is fixed then *the variance of the error is the smaller the sparser the input is*, i.e., the smaller L is. Random mapping will therefore function the better the sparser the data is.

3.2. Random mapping and the SOM

Let us next consider how random mapping of the data vectors affects the further processing of the data. The Self-Organizing Map (SOM) algorithm will serve as an instructive example case since it will be used in the experiments reported in Sec. 4. The conclusions are valid for distance-based clustering algorithms as well.

The SOM algorithm [7] constructs a mapping from the input space onto a usually two-dimensional lattice. Each lattice position called a map unit contains a model vector, and as a result of the algorithm the model vectors of neighboring map units gradually learn to represent similar input vectors. The mapping becomes ordered. The resulting map is an intuitive, abstract representation of the data set. The map can be used for example in data exploration applications, but in a multitude of other applications as well (cf. [7]).

The SOM algorithm consists of two steps that are applied iteratively. First the winning unit whose model vector is closest to the current input is selected, and thereafter the model vectors of the units that are neighbors of the winning unit on the map lattice are updated.

It may be useful to notice that since the random mapping operation is linear, small neighborhoods in the original space will be mapped onto small neighborhoods in the smaller-dimensional space. In the SOM the model vectors of neighboring units are generally close-by, and therefore small neighborhoods in the original space will mostly become mapped onto a single map unit or onto a set of neighboring map units. The SOM will thus probably not be too sensitive to the distortions of the similarities caused by the random mapping.

Before considering the effects the random mapping of the inputs has on the learning of the SOM we must consider the concept of the nullspace of the mapping operator R. The mapping operation can be considered as a "change of basis" to the (non-orthonormal) "basis" formed of the *rows* of R. The rows form a set of random vectors in the original space. The *nullspace* of the operator R is that subspace of the original space that becomes mapped to the zero vector.

Each input vector $\mathbf{n}$ that resides in the original data space can be decomposed into a unique sum of two orthogonal components $\hat{\mathbf{n}}$ and $\tilde{\mathbf{n}} = \mathbf{n} - \hat{\mathbf{n}}$, where $\tilde{\mathbf{n}}$ belongs to the nullspace of R and $\hat{\mathbf{n}}$ to its complement. When the input vector $\mathbf{n}$ is mapped with the random mapping operator the result reflects only the parts of $\mathbf{n}$ that are orthogonal to the nullspace,

$$R\mathbf{n} = R\hat{\mathbf{n}}\,. \tag{11}$$

The projection thus in effect removes the parts of $\mathbf{n}$ that reside in the nullspace of R.

When the mapped vector $R\mathbf{n}(t)$ is input to the SOM at time step t the model vectors $\mathbf{m}_i$ are updated according to the rule

$$\mathbf{m}_i(t+1) = \mathbf{m}_i(t) + h_{ci}(t)[R\mathbf{n} - \mathbf{m}_i(t)] , \qquad (12)$$

where h_{ci} is the so-called neighborhood kernel, a decreasing function of the distance between the units i and c on the map lattice. Here c is the index of the unit whose model vector is closest to $R\mathbf{n}(t)$.

The update in (12) occurs in the *mapped space* but actually we are more interested in comparing the results of the update with the results obtained with a SOM that would operate on the inputs $\mathbf{n}$ in the *original space*. It is in fact possible to consider a "virtual image" of the model vectors $\mathbf{m}_i$ in the original space or, stated more exactly, the virtual images of the model vectors in the complement of the nullspace of the mapping operator R.

If we denote the pseudoinverse of R by $R^\dagger$ then the virtual image of the model vector $\mathbf{m}_i$ in the original space is defined to be $R^\dagger\mathbf{m}_i$. Let us denote this virtual image by $\hat{\mathbf{m}}_i$; the image is the vector that has the smallest norm among all the vectors that R maps onto $\mathbf{m}_i$. If we multiply both sides of (12) by $R^\dagger$ we get

$$\hat{\mathbf{m}}_i(t+1) = \hat{\mathbf{m}}_i(t) + h_{ci}(t)[\hat{\mathbf{n}} - \hat{\mathbf{m}}_i(t)] . \qquad (13)$$

The learning rule then, in effect, corresponds to learning in the original data space, but in the complement of the nullspace of R.

It may seem disadvantageous to deliberately neglect the rest of the vectors $\mathbf{n}$, namely the component $\tilde{\mathbf{n}}$, but it will be demonstrated empirically in Sec. 4 that even a reduction from a 5781-dimensional space to a 90-dimensional one with random mapping produces satisfactory results. It may be striking that in this case the null-space is 5691-dimensional and only 90 *randomly chosen* dimensions are taken into account. The reason for the good results is probably most clearly recognizable based on equation (10): for 90-dimensional vectors the variance in the similarity is smaller than 2.2 % of the largest possible similarity.

4. Experiments: mapping of textual documents in the WEBSOM system

4.1. The WEBSOM system

The WEBSOM [4, 6, 8, 9] is a method for organizing textual documents onto a two-dimensional map display. Nearby locations on the display contain similar documents, which aids in browsing the document collection. The map can also be used for content-addressable search, and for filtering interesting documents from an incoming document stream.

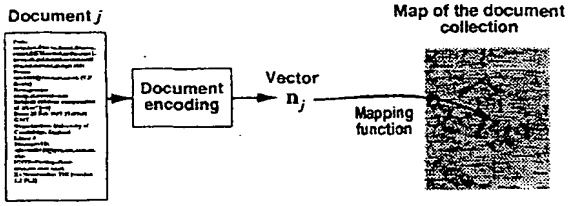

Figure 2. A schematic diagram of the basic building blocks of the WEBSOM system. In the WEBSOM the documents are first encoded into numerical vectors and then mapped onto a two-dimensional display of the document collection using the SOM algorithm.

The task of encoding the documents in the WEBSOM system (Fig. 2) will be used here as a case study of the random mapping method. It should be noted, however, that for the sake of clarity the case study does not include all of the possible ingredients of the WEBSOM system.

4.2. Encoding of documents using random mapping

In a simple but yet very effective document encoding method, called the vector space model [11], the documents are represented by vectors in a space where each dimension corresponds to one word. The value of each component is equal to the relative frequency of occurrence of the corresponding word in the document. Alternatively, some function of the frequency of occurrence and the importance of the word may be used. The resulting vectors can be thought of as representing the *word histograms* of the documents. When the length of the document vector is normalized the direction of the vector will reflect the contents of the document.

It is unfortunately impossible to use the vector space model as such for large document collections since the dimensionality of the resulting document vectors would be very high. There are as many dimensions in the vectors as there are words in the vocabulary. The vector space model thus seems to be an ideal candidate for random mapping. In fact, random mapping of the word histograms has already been shown to produce promising results in preliminary experiments [5].

4.3. Results

The usefulness of the random mapping method in reducing the dimensionality of the document vectors was

measured using an index that has been designed to measure the relative goodness of different document encoding methods in the WEBSOM system. The goal of the WEBSOM is to produce a map where each location contains a set of similar articles and close-by locations contain similar sets. It would be very laborious to assess the success of the method in the subtle details of this task but it is possible to *measure how well different topic areas are separated* on the map of the document collection. The document collection used in the experiments consisted of about 18000 articles from 20 Usenet newsgroups, and the groups were considered to represent different topic areas. It should be noted that although some of the most similar newsgroups were grouped together the groups are still highly overlapping. Thus, the separability of the groups can only be used as a *relative* criterion for comparing different document encoding methods and not as an absolute measure of the goodness of the WEBSOM method.

Before constructing the word histograms for the documents the rarest and some common words were removed. After the removals the dimensionality of the document vectors was 5781. In the histograms each word was weighted with an entropy-based weight [8]. The separability of the newsgroups on the document map was measured by teaching a 768-unit SOM using the encoded documents as inputs, and labeling each map unit according to the group that dominated the unit. The separability of the newsgroups was measured as the total number of documents from the other groups than the dominating one in the nodes. All computations were made using the same text document material; this corresponds to the usage of the WEBSOM method in many real situations. It is often more important to construct a good map of a certain document collection than to be able to generalize the result to new documents.

The separability of the newsgroups as a function of the dimensionality d obtained by the random mapping method is depicted in Figure 3 together with the results obtained with PCA. The PCA is essentially equivalent with the latent semantic indexing method [1] that has been used to reduce the dimensionality of document vectors. The separability obtained with PCA rises very rapidly and saturates around $d \geq 50$. The random mapping requires somewhat larger dimensionalities but if $d \geq 90$ the results are essentially as good as those obtained with PCA, and almost as good as the results obtained with the original vectors. Moreover, the computational complexity of forming the random matrix, $\mathcal{O}(Nd)$, is negligible to the computational complexity of estimating the principal components, $\mathcal{O}(nN^2) + \mathcal{O}(N^3)$ [2]. Here N and d are the dimensionalities before and after the random mapping, respectively, and n is the number of data vectors.

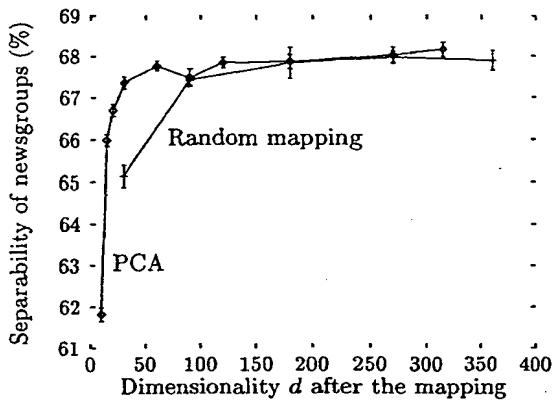

Figure 3. Separability of topic areas on a WEBSOM document map as a function of the dimensionality d of the document vectors obtained by random mapping or PCA. The bars denote the standard deviations of 7 experiments. The separability obtained using the original document vectors was 68.0%.

5. Discussion

The random mapping method has been shown to offer a promising, computationally feasible alternative for dimensionality reduction in situations where the reduced-dimensional data vectors are used for clustering or other similar approaches. Especially if the original dimensionality of the data is very large it is infeasible to use more computationally costly methods like the PCA.

The method has been applied in the WEBSOM document organization system. The dimensionality of the original data vectors that describe the documents is very high, of the order of thousands, and therefore the computations required to construct a Self-Organized Map would be infeasible without a rapid dimensionality reduction method. The random mapping method was demonstrated to produce essentially as good results as the PCA or the original data vectors if the dimensionality of the mapped vectors is about a hundred or more.

The random mapping method has also produced better separability of different topic areas (68% vs. 63%, [5]) than an alternative method [4, 6, 8, 9] in which, however, the encoding of the documents is faster.

There exist straightforward neural implementations of the random mapping method; in this paper the emphasis has, however, been more on the properties of the

mapping than on the implementation.

Acknowledgments

I wish to thank Prof. Teuvo Kohonen for useful discussions concerning the methodology; his ideas have contributed significantly to the present results. I am also grateful to Mr. Jarkko Salojärvi for carrying out the simulations needed for Fig. 3 and to Dr. Timo Honkela for the simulations needed for Fig. 1.

Appendix: The proofs

A. Equation (7)

The distribution of ϵ_{ij} can be derived fairly easily since the vectors $\mathbf{r}_i$ and $\mathbf{r}_j$ in (5) consist of independent normally distributed values that have been normalized so that the length of both of the vectors equals unity. The inner product in (5) is then in fact an estimate of the correlation coefficient between two independent, identically and normally distributed random variables. The normalization of the vectors corresponds to the normalization of the estimate by square roots of the sums of squares of the instances of the random variables. It is an old result, due to Fisher, that $1/2 \ln(1+\epsilon_{ij})/(1-\epsilon_{ij})$ is normally distributed with variance equal to $1/(d-3)$ if d is the number of samples in the estimate. If this equation is linearized around zero the claim follows for large d.

B. Equation (9)

Based on the definition of δ,

$$\begin{aligned}\sigma_\delta^2 &= E[(\sum_{k\neq l} \epsilon_{kl} n_k m_l)(\sum_{p\neq q} \epsilon_{pq} n_p m_q)] \\ &= \sum_{k\neq l}\sum_{p\neq q} n_k m_l n_p m_q E[\epsilon_{kl}\epsilon_{pq}] \, .\end{aligned}$$

It is straightforward to verify that $E[\epsilon_{kl}\epsilon_{pq}] = 0$ unless $k = p$ and $l = q$, or $k = q$ and $l = p$. Hence,

$$\begin{aligned}\sigma_\delta^2 &= \sum_{k\neq l} n_k^2 m_l^2 \sigma_\epsilon^2 + \sum_{k\neq l} n_k m_l n_l m_k \sigma_\epsilon^2 \\ &= (\sum_k n_k^2 \sum_{l\neq k} m_l^2 + \sum_k n_k m_k \sum_{l\neq k} n_l m_l)\sigma_\epsilon^2 \\ &= [1 - \sum_k n_k^2 m_k^2 \\ &\quad +(\sum_k n_k m_k)^2 - \sum_k n_k^2 m_k^2]\sigma_\epsilon^2 \\ &= [1 + (\sum_k n_k m_k)^2 - 2\sum_k n_k^2 m_k^2]\sigma_\epsilon^2 \, .\end{aligned}$$

Here we have used the assumption that the data vectors have been normalized, i.e., $\sum_k n_k^2 = 1$ and $\sum_k m_k^2 = 1$.

References

[1] S. Deerwester, S.T. Dumais, G.W. Furnas, and T.K. Landauer, "Indexing by latent semantic analysis," *Journal of the American Society for Information Science*, Vol. 41, 1990, pp. 391–407.

[2] G.H. Golub and C.F. van Loan, *Matrix Computations*, North Oxford Academic, Oxford, England, 1983.

[3] R. Hecht-Nielsen, "Context vectors: general purpose approximate meaning representations self-organized from raw data", *Computational Intelligence: Imitating Life*, J.M. Zurada, R.J. Marks II, and C.J. Robinson, eds., IEEE Press, Piscataway, NJ, 1994, pp. 43–56.

[4] T. Honkela, S. Kaski, K. Lagus, and T. Kohonen, *Newsgroup exploration with WEBSOM method and browsing interface*, Tech. Report A32, Laboratory of Computer and Information Science, Helsinki University of Technology, Espoo, Finland, 1996.

[5] S. Kaski, "Data exploration using self-organizing maps," *Acta Polytechnica Scandinavica, Mathematics, Computing and Management in Engineering Series*, No. 82, March 1997, Dr.Tech. Thesis, Helsinki University of Technology, Finland.

[6] S. Kaski, T. Honkela, K. Lagus, and T. Kohonen, "Creating an order in digital libraries with self-organizing maps," *Proceedings of WCNN'96, World Congress on Neural Networks*, Lawrence Erlbaum and INNS Press, Mahwah, NJ, 1996, pp. 814–817.

[7] T. Kohonen, *Self-Organizing Maps*, Springer, Berlin, 1995 (Second, extended edition 1997).

[8] T. Kohonen, S. Kaski, K. Lagus, and T. Honkela, "Very large two-level SOM for the browsing of newsgroups," *Proceedings of ICANN96, International Conference on Artificial Neural Networks*, C. von der Malsburg, W. von Seelen, J.C. Vorbrüggen, and B. Sendhoff, eds., Lecture Notes in Computer Science, Vol. 1112, Springer, Berlin, 1996, pp. 269–274.

[9] K. Lagus, T. Honkela, S. Kaski, and T. Kohonen, "Self-organizing maps of document collections: a new approach to interactive exploration," *Proceedings of the Second International Conference on Knowledge Discovery and Data Mining*, E. Simoudis, J. Han, and U. Fayyad, eds., AAAI Press, Menlo Park, California, 1996, pp. 238–243.

[10] H. Ritter and T. Kohonen, "Self-organizing semantic maps," *Biological Cybernetics*, Vol. 61, 1989, pp. 241–254.

[11] G. Salton and M.J. McGill, *Introduction to modern information retrieval*. McGraw-Hill, New York, 1983.